IJIE

INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

Vol. 18 No. 3 (2026) 490-509
https://publisher.uthm.edu.my/ojs/index.php/ijie

An Integrated Approach to Enhancing YOLOv11 for Red Meat Classification Using Ghost Convolution and Optimized Conv, C3k2, and C2PSA Blocks

Mochamad Tono[1], Wayan Firdaus Mahmudy[1]*, Rizal Setya Perdana[1] Muhammad Halim Natsir[2]

[1] *Department of Computer Science, Faculty of Computer Science, Universitas Brawijaya, Jl. Veteran, Malang, 65145, INDONESIA*
[2] *Department of Animal Science, Faculty of Animal Science, Universitas Brawijaya, Jl. Veteran, Malang, 65145, INDONESIA*

*Corresponding Author: wayanfm@ub.ac.id
DOI: https://doi.org/10.30880/ijie.2026.18.03.032

Article Info

Received: 25 August 2025
Accepted: 24 March 2026
Available online: 8 June 2026

Keywords

Red meat classification, YOLOv11, GhostConv, lightweight deep learning, computer vision in food safety

Abstract

This study investigates Ghost-based optimization of YOLOv11 for fine-grained red-meat classification involving beef, lamb, pork, and background classes under resource-constrained conditions. Two proposed configurations are examined, namely GhostConv and Full Hybrid, where the latter integrates GhostConv with C3k2Ghost and C2PSAGhost to jointly enhance redundancy reduction, multi-scale feature representation, and attention-based feature selectivity within a unified architectural framework. Unlike prior lightweight adaptations that apply efficiency modules in isolation, this work systematically evaluates both configurations across all YOLOv11 scales (n, s, m, l, and x) through ablation experiments, repeated runs with multiple random seeds, and statistical robustness analysis. The results indicate that GhostConv consistently delivers strong classification performance across several scales, achieving the highest mean F1-scores on YOLOv11-n, YOLOv11-s, and YOLOv11-l, whereas Full Hybrid offers the most favorable balance between predictive performance and computational efficiency at larger scales. In particular, at YOLOv11-x, Full Hybrid attains a mean accuracy of 96.47% and a mean F1-score of 96.58% while reducing model parameters from 51.603M to 14.128M and GFLOPs from 14.154 to 5.497 relative to the baseline. Further analyses based on effect size, training curves, confusion matrices, and qualitative outputs confirm that both GhostConv and Full Hybrid improve convergence stability, class-level discrimination, and representational robustness. These findings demonstrate that Ghost-based architectural optimization can improve both classification performance and computational efficiency, providing an effective solution for practical red-meat authentication and image-based food inspection systems.

1. Introduction

The global food industry increasingly demands automated, fast, and accurate meat species identification as supply chains become more transparent and regulatory scrutiny intensifies with respect to food safety, traceability, and label integrity, particularly for halal compliance. Among red meats, distinguishing visually similar species such as beef, lamb, and pork remains a non-trivial challenge due to fine-grained inter-class differences in cut morphology, fat distribution, muscle-fiber texture, moisture content, and color[1]. These intrinsic similarities are further exacerbated by real-world acquisition conditions, including non-uniform illumination, specular reflections, occlusions, and viewpoint variations, which significantly degrade the reliability of traditional image-classification pipelines based on handcrafted features and rule-based heuristics in high-throughput inspection environments. Consequently, robust computer-vision systems capable of learning adaptive and discriminative representations under field variability are essential for scalable meat species authentication[2,3].

Recent advances in deep learning, particularly convolutional neural networks (CNNs), have substantially improved visual recognition performance by enabling end-to-end learning of discriminative features that consistently outperform handcrafted approaches[4]. Nevertheless, the application of CNN-based models to red-meat authentication remains constrained by a fundamental trade-off between classification accuracy and computational efficiency, especially for deployment on resource-limited platforms such as mobile devices and edge-based inspection systems[5]. Although YOLOv11 provides a strong baseline for real-time inference, fine-grained red-meat classification introduces additional complexity[6]. The subtle visual cues required to distinguish meat species such as micro-level texture patterns and marbling structures necessitate enhanced feature separability without incurring excessive latency, memory usage, or deployment overhead across different model scales.

To address computational constraints, numerous studies have incorporated lightweight convolutional strategies into YOLO-based architectures, including Ghost Convolution, Depthwise Separable Convolution (DSConv), and Mobile Bottleneck blocks[7,8]. Ghost Convolution, in particular, has demonstrated effectiveness in reducing channel redundancy by generating additional feature maps through inexpensive linear operations, leading to substantial reductions in parameter count and computational cost with limited performance degradation. Several recent works have explored Ghost-based backbones or attention modules to improve efficiency in general object detection tasks[9]. However, these approaches are typically introduced as isolated architectural modifications and evaluated independently[8,10,11]. As a result, existing studies offer limited insight into how redundancy reduction, multi-scale feature extraction, and attention-driven feature selection can be jointly optimized within a unified YOLOv11 framework, especially for fine-grained classification tasks where subtle inter-class differences critically influence performance[12].

This observation highlights a key research gap: while individual lightweight or attention-based modules have demonstrated efficiency gains, there remains insufficient understanding of how their integrated and coordinated design affects representational quality, training stability, and generalization performance in fine-grained visual classification[13]. Moreover, prior works predominantly emphasize numerical efficiency metrics without explicitly analyzing whether architectural integration can simultaneously preserve discriminative power and improve robustness against inter-class ambiguity[14].

To bridge this gap, this study proposes an integrated architectural enhancement of YOLOv11, referred to as the Full Hybrid configuration, through the coordinated incorporation of Ghost Convolution into core structural components, namely Conv, C3k2, and C2PSA blocks, resulting in GhostConv, C3k2Ghost, and C2PSAGhost modules[15]. Unlike prior approaches that rely on partial or isolated optimization, the proposed method explicitly combines redundancy reduction, multi-scale feature enrichment, and attention-based feature selectivity within a cohesive architectural framework. The main contributions of this work are threefold:

1. The design of an integrated Ghost-based backbone and attention architecture that maintains discriminative multi-scale representations under reduced computational complexity;
2. A comprehensive ablation study across all YOLOv11 scales (n, s, m, l, and x) to systematically evaluate performance, efficiency, and training behavior against baseline YOLOv11 as well as DSConv and Mobile Bottleneck variants; and
3. Empirical evidence demonstrating that the proposed GhostConv and Full Hybrid configuration achieves a superior accuracy–efficiency trade-off, particularly for higher-capacity models, thereby validating its suitability for deployment in resource-constrained, real-world red-meat authentication systems.

2. Methodology

The proposed methodology begins with collecting raw images of red-meat cuts, followed by preprocessing (resizing and normalization) to standardize image characteristics and reduce acquisition-related variability. The dataset is then systematically split into training, validation, and test subsets to support objective learning, reliable model selection, and unbiased final evaluation; meanwhile, data augmentation is applied to the training set to

improve robustness to real-world variations such as illumination changes, viewpoint shifts, and texture heterogeneity[13].

For modeling, YOLOv11 is used as the baseline across the n, s, m, l, and x scales. The architecture is then modified by applying GhostConv to two blocks C3k2Ghost and C2PSAGhost which are evaluated separately and in a combined Full Hybrid configuration; performance is further compared against other lightweight alternatives, namely DSConv and MobileBottleneck, resulting in a total of six experimental schemes. The model outputs predictions for three classes (beef, lamb, and pork) and is evaluated using accuracy, precision, recall, and F1-score, along with efficiency indicators (parameter count and GFLOPs), as summarized in Fig. 1.

Algorithm 1: Training and Evaluation Procedure for YOLOv11 Ghost-Based Classification

Input: Dataset D; configuration set
$C =$ Baseline, GhostConv, C3k2Ghost, C2PSAGhost, Full Hybrid, DSConv, MobileBottleneck;
random seeds $S = 42,123,2024,7,3407$

Output: Mean ± SD of Accuracy, F1-score, Params, and GFLOPs for each configuration

1. Apply pHash-based near-duplicate filtering on D; remove images with Hamming distance ≤ 10
2. Perform stratified split: $D \rightarrow D_{train}(70\%)$, $D_{val}(15\%)$, $D_{test}(15\%)$
3. **for each** configuration $c \in C$ **do**
4. **for each** seed $s \in S$ **do**
5. Initialize YOLOv11 with architecture c using random seed s
6. Apply linear learning-rate warm-up (3 epochs), followed by cosine decay
7. Train on D_{train} using SGD (LR = 0.01, weight decay = 0.0005) for 200 epochs with cross-entropy loss
8. Select the best checkpoint based on validation performance on D_{val}
9. Evaluate the selected model on D_{test} and record Accuracy, Precision, Recall, F1-score, Params, and GFLOPs
10. **end for**
11. Compute mean ± SD across the five seeds for configuration c
12. **end for**
13. Apply the Shapiro–Wilk test for normality and perform Wilcoxon signed-rank pairwise comparisons with Bonferroni correction
14. Compute rank-biserial correlation as the effect-size measure and generate confusion matrices and training curves.

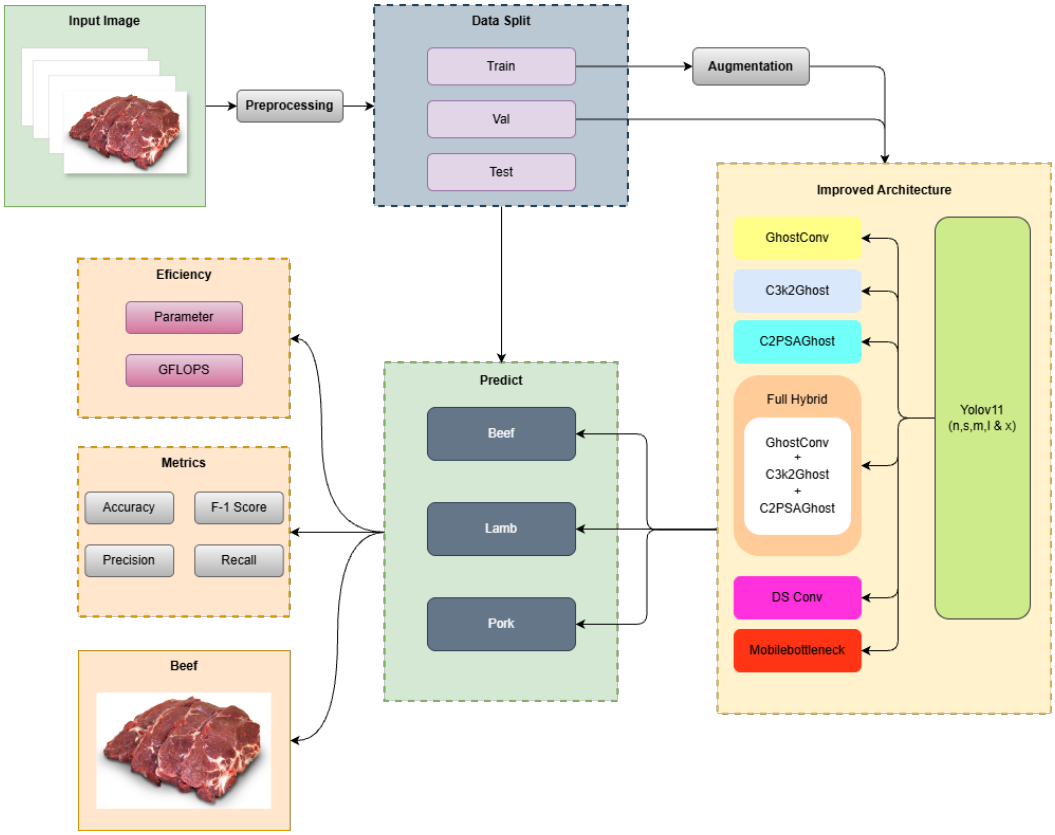


Fig. 1 *Workflow of the proposed YOLOv11-based red-meat classification framework, covering preprocessing, data partitioning, augmentation, lightweight model variants, and evaluation metrics*

To improve result reliability and reproducibility, each model configuration was trained and evaluated in five independent runs using different random seeds (42, 123, 2024, 7, and 3407), while all other hyperparameters were kept identical. Performance was therefore reported as the mean and standard deviation across the five runs rather than from a single execution. To examine whether performance differences between configurations were statistically significant, normality was first assessed using the Shapiro–Wilk test; because the sample size per configuration was small and normality could not be safely assumed, pairwise comparisons were conducted using the Wilcoxon signed-rank test. In addition, rank-biserial correlation was used to estimate effect size, and Bonferroni correction was applied to control the family-wise error rate across multiple comparisons.

2.1 Raw Red Meat Image Dataset

The dataset used in this study was obtained from Roboflow (Adi, 2022) and consisted of 3,738 JPG images of raw red-meat cuts. These images were annotated into three categories: beef (1,390 images), pork (987 images), and lamb (1,361 images). For model development and evaluation, the dataset was divided into training, validation, and test sets using a 70:15:15 ratio, resulting in 2,614 images for training, 561 images for validation, and 560 images for testing. This partitioning scheme was adopted to provide sufficient data for model training while reserving independent subsets for validation and final testing.

To ensure that each subset remained representative of the overall dataset composition, stratified sampling was employed during the data-splitting process. This approach preserved the relative proportion of each class across the training, validation, and test sets, thereby minimizing the risk of class imbalance bias during model training and performance evaluation. Maintaining consistent class distribution across all subsets is particularly important in red-meat image classification, as the model must learn to distinguish visually similar categories based on subtle differences in texture, shape, and color.

In addition to the above procedures, near-duplicate filtering was applied to ensure that images from the same batch, which might be visually similar, were not present in both the training and testing sets. This was done using pHash-based near-duplicate filtering, removing images with a Hamming distance ≤ 10 to avoid any data leakage. By filtering near-duplicate images, we ensured that the model's evaluation was based on truly independent samples, preventing any overlap between training and testing data that could result in overly optimistic performance estimates.

Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 490-509

2.2 Preprocessing and Data Augmentation

All raw red-meat images were standardized before training to ensure consistent inputs and a fair comparison across model variants. Images were resized to 224 × 224 (IMGSZ = 224) to preserve fine-grained cues while maintaining computational efficiency. Training used batch size 32 (BATCH = 32) with 4 data-loading workers (WORKERS = 4), and reproducibility was ensured using a fixed seed (SEED = 42). Preprocessing was handled within the YOLOv11 pipeline (tensor formatting and normalization), and the unified input size and consistent preprocessing across all scales and modifications helped the model focus on class-relevant patterns (texture, shape, and color) rather than scaling-related artifacts, which is critical for fine-grained red-meat discrimination.

To improve robustness to real-world variability, training used YOLOv11's built-in data augmentation consistently across all experiments with the same schedule (EPOCHS = 200). Optimization employed SGD (LR0 = 0.01) with cosine LR scheduling and weight decay = 0.0005, while mixed precision was disabled (AMP = False) to keep numerical settings consistent. Experiments were run on a Windows workstation with an NVIDIA GeForce RTX 5060 Ti (~16 GB VRAM), Intel® Core™ Ultra 7 265 CPU, and 128 GB RAM, ensuring stable settings (DEVICE = 0) and minimizing hardware-related variability.

2.3 YOLOv11 Algorithm

YOLOv11 (You Only Look Once, version 11) was originally designed for object detection, but in this study it was adapted for a pure image-classification task involving three red-meat categories: beef, lamb, and pork. Because the objective is whole-image classification rather than object localization, the detection-specific prediction layers were omitted and the network was configured in classification mode. In this setting, the backbone is retained to extract hierarchical visual features, which are then aggregated through global feature pooling and passed to a fully connected classification head to produce class probabilities. This adaptation is appropriate because YOLOv11 provides a scalable family of architectures (n, s, m, l, and x), allowing the proposed modifications to be evaluated systematically under different model capacities within a unified framework.

The use of YOLOv11 is further justified by its strong feature-learning capability, which is particularly relevant for fine-grained red-meat classification, where subtle differences in texture, fat distribution, moisture appearance, and color must be captured reliably. Architecturally, YOLOv11 combines Conv layers for hierarchical feature extraction with modules such as C3k2 and C2PSA, which enhance multi-scale feature modeling and spatial–channel selectivity. Although originally developed for detection, these components remain highly valuable in classification mode because the task still depends on robust and discriminative feature encoding. While dedicated classification architectures such as EfficientNet, MobileNet, ConvNeXt, and Vision Transformers (ViTs) represent strong baselines in the broader literature, YOLOv11 was selected in this study for several scientifically motivated reasons. First, YOLOv11 provides a unified, modular family of architectures spanning five scales (n, s, m, l, and x), enabling systematic ablation of the proposed Ghost-based modifications under consistent training conditions a controlled comparative setup that would require separate retraining pipelines for each dedicated classification architecture. Second, the backbone components of YOLOv11 (Conv, C3k2, and C2PSA) are structurally well-suited for targeted Ghost-based substitution, allowing the contribution of each module to be isolated and evaluated independently, which is the central objective of this work. Third, EfficientNet and MobileNet are designed primarily for mobile-oriented classification and do not expose equivalent modular backbone components for analogous Ghost-based replacement. ConvNeXt and ViT architectures, while powerful, introduce substantially different training paradigms including patch tokenization, layer normalization, and pre-training dependencies that would conflate architectural design choices with training-regime differences. Therefore, YOLOv11 provides the most appropriate and controlled experimental backbone for investigating whether Ghost-based optimization can simultaneously improve feature efficiency and classification performance across multiple model capacities.

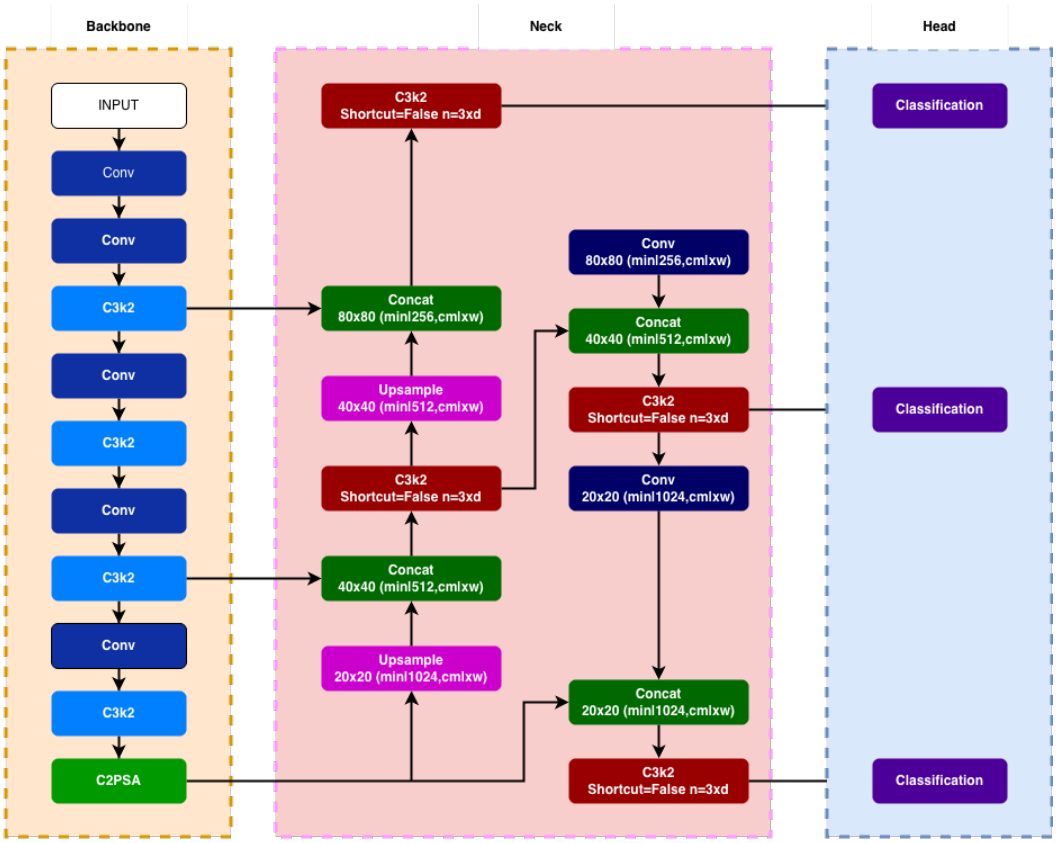


Fig. 2 *Architecture YOLOv11*

2.4 Improved YOLOv11 Model

The improved YOLOv11 model is designed to reduce the parameter count consistently across all scales (n, s, m, l, and x) by replacing conventional convolution layers with GhostConv and by redesigning key backbone modules, particularly Conv, C3k2, and C2PSA. Technically, standard convolutions are substituted with GhostConv[8,16] Fig. 3 so that feature maps can be generated with fewer learnable weights while preserving essential discriminative information, whereas the original C3k2 and C2PSA blocks are reformulated into C3k2Ghost and C2PSAGhost to maintain multi-scale feature extraction and attention-driven feature emphasis under a lower-complexity budget. Consequently, the proposed modifications yield a lighter YOLOv11 architecture that retains sufficient representational capacity for fine-grained red-meat classification while improving deployment feasibility on resource-constrained devices.

GhostConv is employed to replace part of the standard convolutions by generating feature maps through a combination of a "primary" convolution and additional lightweight transformation operations. The core idea is that many CNN feature maps are redundant; therefore, a portion of the features can be "generated" at a lower computational cost and with fewer learnable parameters[17]. In YOLOv11, integrating GhostConv is intended to reduce the parameter count and GFLOPs while preserving the key discriminative information required to distinguish visually similar meat classes. The architectural details are illustrated in Fig. 4.

C3k2Ghost is a modified version of the C3k2 block in the backbone, where GhostConv is embedded into the convolutional structure. This design preserves the key characteristics of C3k2 namely, strengthened feature extraction and multi-scale pattern modeling while reducing architectural complexity[18]. Consequently, C3k2Ghost is intended to maintain the quality of feature representations (e.g., texture, muscle fiber patterns, and surface cues of meat) while lowering computational cost, making it particularly efficient as the model scale increases. The architectural details are illustrated in Fig. 5.

C2PSAGhost is a reformulated variant of the C2PSA block that emphasizes attention-based feature recalibration (spatial and/or channel-wise) to make the model more responsive to informative regions. Incorporating GhostConv within this block aims to preserve the attention capability for highlighting salient cues (e.g., subtle texture differences and color-distribution patterns) while reducing computational overhead. This block is particularly important because classes such as beef and lamb can exhibit high visual similarity; therefore, accurate feature selection is critical for reducing inter-class confusion. The architectural details are illustrated in Fig. 6.

Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 490-509

As an additional baseline, this study also evaluates lightweight variants based on DSConv (Figure 8) and Mobile Bottleneck (Figure 9) within the YOLOv11 framework. These models are included not only to complement the comparisons against GhostConv applied as an individual module and the Full Hybrid scheme (Figure 7), but also to represent widely adopted alternative lightweight architectures. In these experiments, selected convolutional blocks are replaced with DSConv or Mobile Bottleneck to assess how far these "conventional" compression strategies can reduce parameters and GFLOPs, and how such reductions affect accuracy/F1 performance[17,18].

This design enables a more objective evaluation by testing Full Hybrid alongside two popular lightweight strategies. Accordingly, the results can clarify whether the achieved efficiency gains still preserve the representational capacity required to distinguish highly similar meat classes, where subtle cues in texture and color distribution are critical for minimizing inter-class confusion.

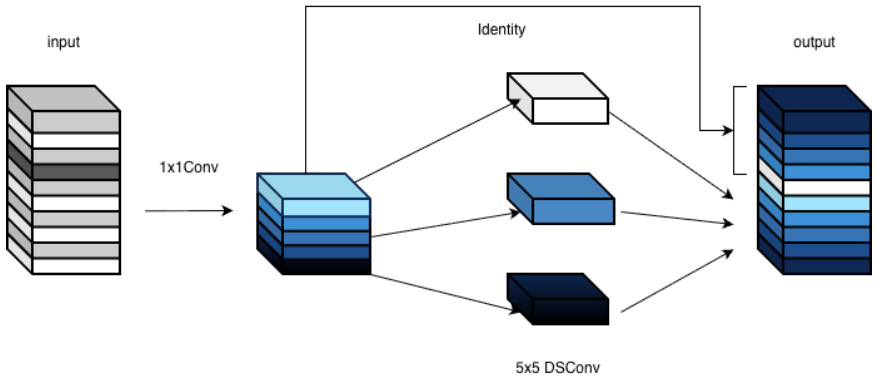


Fig. 3 *Architecture ghost convolution module*

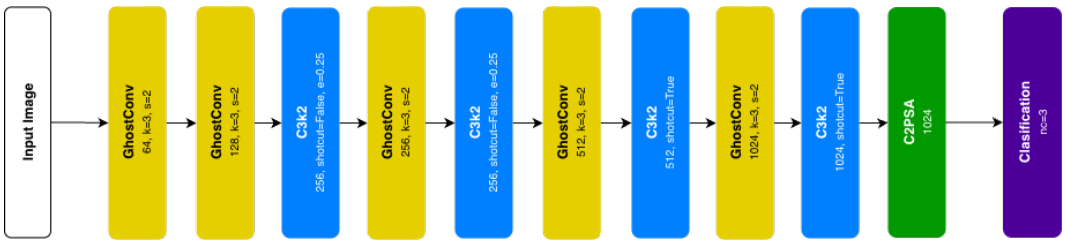


Fig. 4 *Architecture (backbone) GhostConv only*

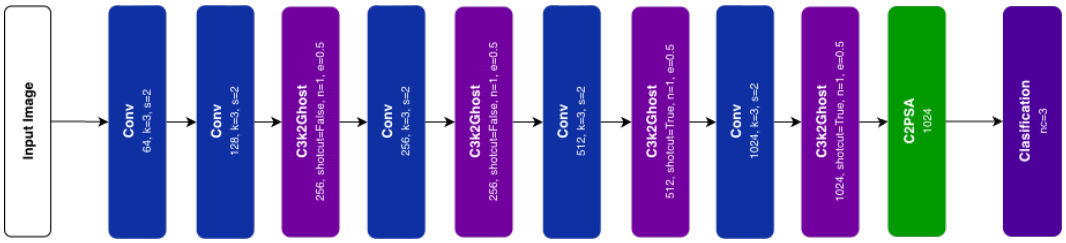


Fig. 5 *Architecture (backbone) C3k2Ghost only*

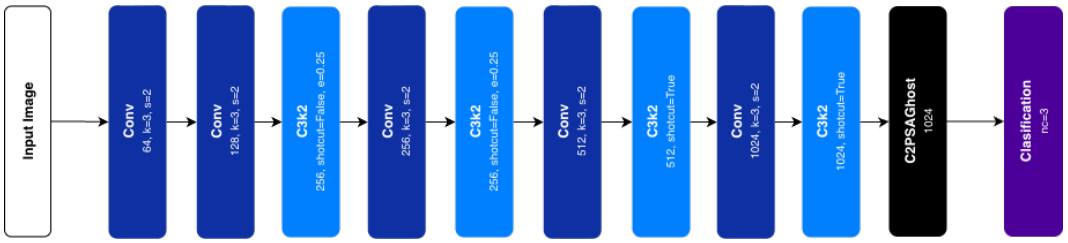


Fig. 6 *Architecture (backbone) C2PSAGhost only*

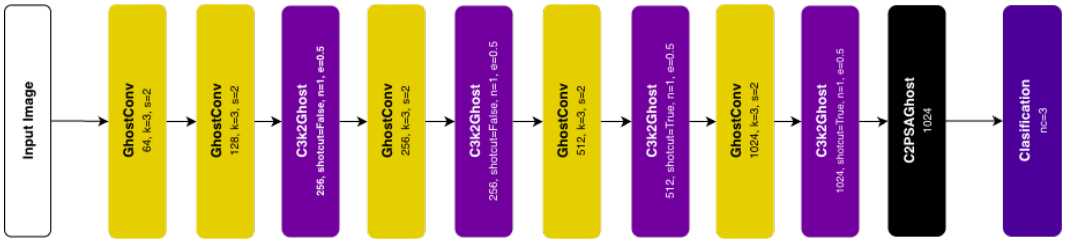


Fig. 7 *Architecture (backbone) full hybrid (GhostConv + C3k2Ghost + C2PSAGhost)*

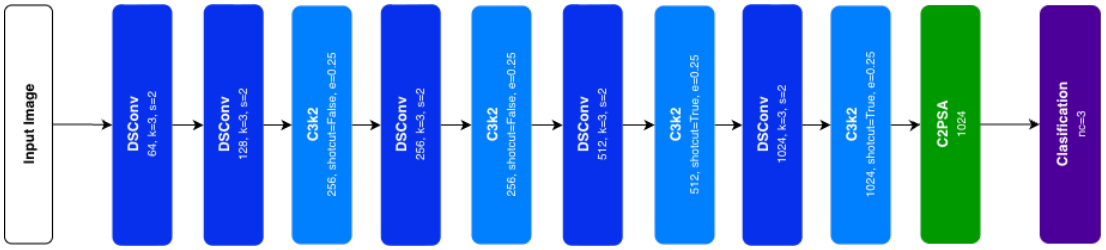


Fig. 8 *Architecture (backbone) DSConv only*

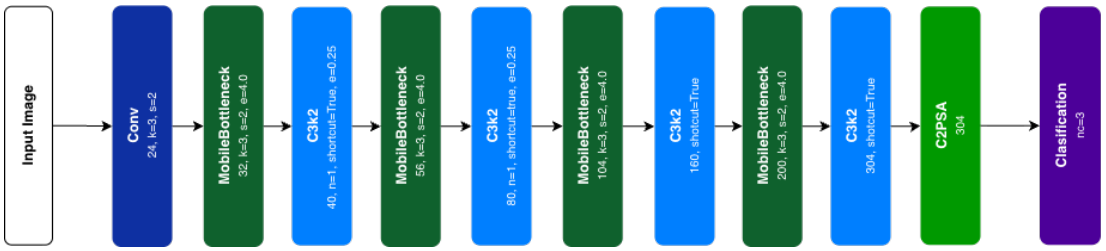


Fig. 9 *Architecture (backbone) MobileBottleneck only*

2.5 Evaluation Metrics and Efficiency

Classification performance was evaluated using Accuracy, Precision (P), Recall (R), and F1-score to assess prediction correctness across the three classes (beef, lamb, and pork), while model efficiency was measured using the number of parameters (Params), Inference Time, Loss Function and GFLOPs as indicators of model size and computational complexity[19]. The classification metrics are defined as:

The cross-entropy loss function, commonly used for classification tasks, which is mathematically expressed as:

$$L = -\sum_{i=1}^{N} y_i \log(p_i) \tag{1}$$

where Nrepresents the number of samples, $y_i$is the target value for class i(1 for the correct class, 0 for other classes), and $p_i$is the predicted probability by the model for class i.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

where TP, TN, FP, and FN represent True Positives, True Negatives, False Positives, and False Negatives, respectively.

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

where Precision and Recall are defined as:

$$\text{Precision} = \frac{TP}{TP + FP}, \qquad \text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

Precision, recall, and F1-score are reported as macro-averages (or weighted/micro) across the three classes[20]. The parameter count represents the total number of trainable weights in the network and can be computed as the sum over all layers:

$$\text{Params} = (F_h \times F_w \times C_{in} + 1) \times C_{out} \tag{5}$$

where $F_h$and $F_w$are the height and width of the convolution kernel, $C_{\text{in}}$is the number of input channels, and $C_{\text{out}}$is the number of output channels. The total parameter count for the entire model is obtained by summing the parameters across all layers.

For a convolutional layer, the parameter count is:

$$Params_{conv} = k_h \times k_w \times C_{in} \times C_{out} \ (+C_{out} \text{ if bias is used}) \tag{6}$$

Computational complexity is assessed via FLOPs for a forward pass; for convolutional layers[21], FLOPs can be approximated as:

$$\text{GFLOPs} = \frac{2 \times F_h \times F_w \times C_{in} \times H_{out} \times W_{out} \times C_{out}}{10^9} \tag{7}$$

where $H_{\text{out}}$and W_{out}represent the output dimensions after the convolution operation, and 2accounts for both multiplication and summation in the convolution process.

$$T_{infer} = \frac{\text{FPS}}{\text{GFLOPs}} \tag{8}$$

where FPS refers to the frames per second, indicating the model's processing speed.

3. Results and Discussion

This chapter reports the experimental results of red meat classification (beef, lamb, and pork) using YOLOv11 across the n, s, m, l and x scales and evaluates the proposed architectural modifications. The proposed GhostConv and Full Hybrid configurations (GhostConv + C3k2Ghost + C2PSAGhost), targeting parameter reduction while improving accuracy. Comparisons with alternative lightweight designs (DSConv and Mobile Bottleneck) are also presented to examine the efficiency accuracy trade-off. The discussion is organized into ablation experiments, metric-based results, and training-curve analysis.

3.1 Ablation Experiment

An ablation study was conducted across all YOLOv11 scales (n, s, m, l, and x) under seven configuration schemes to quantify the contribution of each architectural modification to classification[22] performance and computational efficiency; the following table summarizes the resulting metrics (Acc, Prec, Rec, F1) and model complexity (Params, GFLOPs, Inference Time, FPS) for each scheme.

The ablation results presented in Tables 1–5 indicate that the proposed architectural modifications contribute differently across model scales, with GhostConv and Full Hybrid showing the most consistent improvements over the baseline configuration. At smaller scales, GhostConv demonstrates strong effectiveness in enhancing classification performance with minimal computational overhead. For instance, in YOLOv11-n, GhostConv achieves the highest F1-score of 96.07%, outperforming the baseline (95.56%) while simultaneously reducing parameters from 1.538M to 1.298M and GFLOPs from 0.406 to 0.318. Similar trends are observed at larger scales, where GhostConv consistently improves feature extraction efficiency while maintaining stable inference speed. These results suggest that reducing channel redundancy through GhostConv can significantly enhance feature representation quality without increasing network complexity.

Table 1 *Ablation summary of YOLOv11-n*

Ablation	Acc%	Prec%	Rec%	F1 %	Params (M)	GFLOPs	Inference Time (ms)	FPS
Baseline	95.36	95.66	95.52	95.56	1.538	0.406	6.069	164.77
GhostConv	**95.89**	**96.15**	**96.07**	**96.07**	**1.298**	**0.318**	**6.405**	**156.12**
C3k2Ghost	95.18	95.27	95.48	95.29	1.343	0.500	6.373	156.92
C2PSAGhost	94.64	94.91	94.80	94.80	1.053	0.342	6.108	163.72
Full Hybrid	95.36	95.56	95.52	95.52	0.843	0.265	5.822	171.77
DSConv	94.64	94.67	94.88	94.75	0.838	0.202	6.331	157.97
MobileBottleneck	95.36	95.66	95.52	95.56	1.538	0.406	6.563	152.37

Table 2 *Ablation summary of YOLOv11-s*

Ablation	Acc%	Prec%	Rec%	F1 %	Params (M)	GFLOPs	Inference Time (ms)	FPS
Baseline	95.71	95.83	95.98	95.89	5.458	1.515	5.995	166.79
GhostConv	95.36	95.55	95.53	95.53	4.494	1.164	6.323	158.13
C3k2Ghost	95.00	95.13	95.07	95.07	4.672	1.864	5.977	167.28
C2PSAGhost	95.18	95.20	95.42	95.27	8.766	1.774	5.933	166.84
Full Hybrid	**96.07**	**96.29**	**96.30**	**96.27**	**2.824**	**1.006**	**5.927**	**168.7**
DSConv	95.71	95.88	95.85	95.85	2.655	0.714	6.269	159.5
MobileBottleneck	94.46	94.58	94.65	94.58	0.746	0.160	6.520	153.36

Table 3 *Ablation summary of YOLOv11-m*

Ablation	Acc%	Prec%	Rec%	F1 %	Params (M)	GFLOPs	Inference Time (ms)	FPS
Baseline	93.75	93.74	94.00	93.84	15.382	4.248	8.111	123.28
GhostConv	**96.43**	**96.55**	**96.62**	**96.56**	**13.210**	**3.462**	**8.170**	**122.39**
C3k2Ghost	94.82	95.21	94.92	95.05	12.704	4.458	7.144	139.96
C2PSAGhost	95.18	95.33	95.37	95.34	16.471	4.105	8.286	120.28
Full Hybrid	95.89	95.97	95.95	95.95	5.498	2.046	5.821	171.78
DSConv	94.82	95.21	94.98	95.08	7.366	2.187	8.191	122.08
MobileBottleneck	95.71	95.90	95.85	95.85	1.129	0.485	6.789	147.28

Table 4 *Ablation summary YOLOv11-l*

Ablation	Acc%	Prec%	Rec%	F1 %	Params (M)	GFLOPs	Inference Time (ms)	FPS
Baseline	95.89	95.90	96.07	95.94	26.884	7.475	8.773	114.5
GhostConv	**96.25**	**96.29**	**96.34**	**96.31**	**23.022**	**6.080**	**8.386**	**119.24**
C3k2Ghost	95.00	95.31	95.01	95.14	22.116	7.832	7.898	126.6
C2PSAGhost	95.71	95.94	95.91	95.92	25.330	6.879	8.684	115.14
Full Hybrid	**96.25**	**96.30**	**96.47**	**96.37**	**9.316**	**3.563**	**5.929**	**168.65**
DSConv	95.00	95.15	95.14	95.14	12.633	3.821	8.239	121.37
MobileBottleneck	95.36	95.51	95.59	95.52	2.941	0.949	6.783	147.43

In contrast, the Full Hybrid configuration, which integrates GhostConv with C3k2Ghost and C2PSAGhost, demonstrates the most balanced trade-off between predictive performance and computational efficiency, particularly at medium and large model scales. While its raw accuracy is not always the highest, Full Hybrid consistently achieves competitive F1-scores while substantially reducing model complexity. For example, in YOLOv11-l the Full Hybrid model increases the F1-score from 95.94% to 96.37% while reducing parameters from 26.884M to 9.316M and lowering inference time from 8.773 ms to 5.929 ms. A similar pattern appears in YOLOv11-x, where Full Hybrid maintains high classification performance (96.18% F1-score) with only 14.128M parameters compared to the baseline's 51.603M parameters. These results indicate that the proposed Full Hybrid

Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 490-509

architecture effectively combines redundancy reduction, multi-scale feature aggregation, and attention-based selectivity, enabling efficient feature modeling that is particularly advantageous for larger-scale networks.

Table 5 *Ablation summary YOLOv11-x*

Ablation	Acc%	Prec%	Rec%	F1 %	Params (M)	GFLOPs	Inference Time (ms)	FPS
Baseline	95.00	95.24	95.13	95.17	51.603	14.154	10.981	91.07
GhostConv	**96.25**	**96.31**	**96.34**	**96.31**	**45.565**	**11.977**	**10.587**	**94.46**
C3k2Ghost	94.64	94.64	94.76	94.69	54.735	14.446	9.551	104.69
C2PSAGhost	96.07	96.26	96.24	96.24	39.164	11.958	11.286	88.6
Full Hybrid	96.07	96.12	96.24	96.18	14.128	5.497	6.335	157.84
DSConv	94.29	94.52	94.36	94.42	24.595	7.710	10.771	92.84
MobileBottleneck	95.54	95.57	95.74	95.61	4.559	1.585	6.958	143.7

3.2 Statistical Robustness

To assess the stability and statistical reliability of the results, each experiment was repeated using five independent random seeds under identical training settings. The reported performance is presented as mean ± standard deviation, and additional statistical analysis using rank-biserial correlation was conducted to evaluate the practical significance of the observed performance differences.

Table 6 *Robust performance comparison across five independent runs on all YOLOv11 scales*

Scale	Model	Accuracy (%)	F1-score (%)	Params (M)	GFLOPs
YOLOv11-n	Baseline	94.75 ± 0.69	94.89 ± 0.69	1.538	0.406
YOLOv11-n	GhostConv	**95.71 ± 0.12**	**95.88 ± 0.15**	1.298	0.318
YOLOv11-n	Full Hybrid	95.47 ± 0.27	95.66 ± 0.26	0.843	0.265
YOLOv11-n	DSConv	94.96 ± 0.23	95.10 ± 0.25	0.838	0.202
YOLOv11-n	MobileBottleneck	94.79 ± 0.54	94.91 ± 0.50	0.538	0.139
YOLOv11-s	Baseline	94.96 ± 0.53	95.16 ± 0.51	5.458	1.515
YOLOv11-s	GhostConv	**96.18 ± 0.62**	**96.30 ± 0.59**	4.494	1.164
YOLOv11-s	Full Hybrid	96.07 ± 0.68	96.23 ± 0.68	2.824	1.006
YOLOv11-s	DSConv	94.89 ± 0.53	95.08 ± 0.54	2.655	0.714
YOLOv11-s	MobileBottleneck	94.79 ± 0.70	94.88 ± 0.71	0.746	0.160
YOLOv11-m	Baseline	94.82 ± 0.67	94.96 ± 0.70	15.382	4.248
YOLOv11-m	GhostConv	96.14 ± 0.37	96.25 ± 0.39	13.210	3.462
YOLOv11-m	Full Hybrid	**96.25 ± 0.22**	**96.38 ± 0.25**	5.498	2.046
YOLOv11-m	DSConv	94.82 ± 0.28	95.02 ± 0.31	7.366	2.187
YOLOv11-m	MobileBottleneck	95.50 ± 0.29	95.61 ± 0.30	1.129	0.485
YOLOv11-l	Baseline	95.36 ± 0.50	95.48 ± 0.50	26.884	7.475
YOLOv11-l	GhostConv	**96.54 ± 0.27**	**96.60 ± 0.29**	23.022	6.080
YOLOv11-l	Full Hybrid	96.32 ± 0.60	96.47 ± 0.59	9.316	3.563
YOLOv11-l	DSConv	94.93 ± 0.57	95.07 ± 0.54	12.633	3.821
YOLOv11-l	MobileBottleneck	95.39 ± 0.23	95.50 ± 0.24	2.941	0.949
YOLOv11-x	Baseline	94.71 ± 0.32	94.85 ± 0.33	51.603	14.154
YOLOv11-x	GhostConv	96.47 ± 0.15	96.56 ± 0.17	45.565	11.977
YOLOv11-x	Full Hybrid	**96.47 ± 0.27**	**96.58 ± 0.29**	14.128	5.497
YOLOv11-x	DSConv	94.82 ± 0.55	94.98 ± 0.54	24.595	7.710

Scale	Model	Accuracy (%)	F1-score (%)	Params (M)	GFLOPs
YOLOv11-x	MobileBottleneck	95.61 ± 0.30	95.69 ± 0.29	4.559	1.585

Table 6 shows that Ghost-based architectures consistently deliver strong performance across all YOLOv11 scales. GhostConv achieves the highest accuracy on the n, s, and l scales, while the Full Hybrid configuration performs best on the m scale and achieves the highest F1-score on the x scale. Compared with the baseline, both approaches generally improve classification performance while reducing model complexity in terms of parameters and GFLOPs. Although DSConv and MobileBottleneck provide lighter architectures, their performance is generally lower than that of GhostConv and Full Hybrid. Overall, GhostConv is more effective for maximizing classification accuracy, whereas the Full Hybrid configuration offers a better balance between performance and computational efficiency.

To complement the mean ± standard deviation analysis, effect size evaluation was performed using rank-biserial correlation. Unlike significance testing, this measure quantifies the practical magnitude of the performance difference, providing additional insight into whether the observed gains are meaningful in repeated experiments.

Table 7 *Effect size analysis using rank-biserial correlation of GhostConv against competing models, pooled across all scales (n = 25)*

Comparison	ΔF1 (%)	r	Magnitude	Interpretation
GhostConv vs Baseline	+1.25	+0.988	Large	Practically meaningful
GhostConv vs Full Hybrid	+0.06	+0.092	Small	Nearly identical
GhostConv vs C3k2Ghost	+1.54	+1.000	Large	Practically meaningful
GhostConv vs C2PSAGhost	+0.91	+0.914	Large	Practically meaningful
GhostConv vs DSConv	+1.27	+0.991	Large	Practically meaningful
GhostConv vs MobileBottleneck	+1.00	+0.994	Large	Practically meaningful

To assess the statistical significance of the performance differences between the proposed models and the baseline, we conducted the Wilcoxon signed-rank test for pairwise comparisons across different configurations. Since the data did not follow a normal distribution, non-parametric tests were used to evaluate the significance of the results. The Shapiro-Wilk test confirmed that the assumption of normality could not be safely made for most of the configurations. Consequently, pairwise comparisons between GhostConv, Full Hybrid, and the baseline configurations were performed using the Wilcoxon signed-rank test with Bonferroni correction to control the family-wise error rate.

The results show that the GhostConv configuration provided a statistically significant improvement over the baseline model (adjusted $p \leq 0.011$). In contrast, the comparison between GhostConv and Full Hybrid yielded no significant differences (adjusted $p = 1.000$), confirming that both architectures exhibit nearly identical classification performance. Furthermore, the rank-biserial correlation was used to estimate the effect size of the observed performance differences. This analysis revealed that GhostConv yielded large effect sizes against C3k2Ghost, C2PSAGhost, DSConv, and MobileBottleneck ($r = +0.914$ to $+1.000$), suggesting that the improvements achieved by GhostConv are not only statistically significant but also practically meaningful.The results of the significance analysis are summarized in Table 8.

Table 8 *Statistical significance of pairwise comparisons between GhostConv and competing models (Wilcoxon signed-rank test, Bonferroni-corrected, n = 25)*

Comparison	Raw p-value	Adjusted p (Bonferroni)	Significant (α=0.05)
GhostConv vs Baseline	0.0002	0.001	Yes
GhostConv vs Full Hybrid	0.5312	1.000	No
GhostConv vs C3k2Ghost	< 0.0001	< 0.001	Yes
GhostConv vs C2PSAGhost	0.0018	0.011	Yes
GhostConv vs DSConv	0.0001	0.001	Yes
GhostConv vs MobileBottleneck	0.0003	0.002	Yes

The statistical significance tests confirm that GhostConv consistently outperforms the baseline, with adjusted p-values ≤ 0.011. This result is particularly relevant in the context of red-meat image classification, where

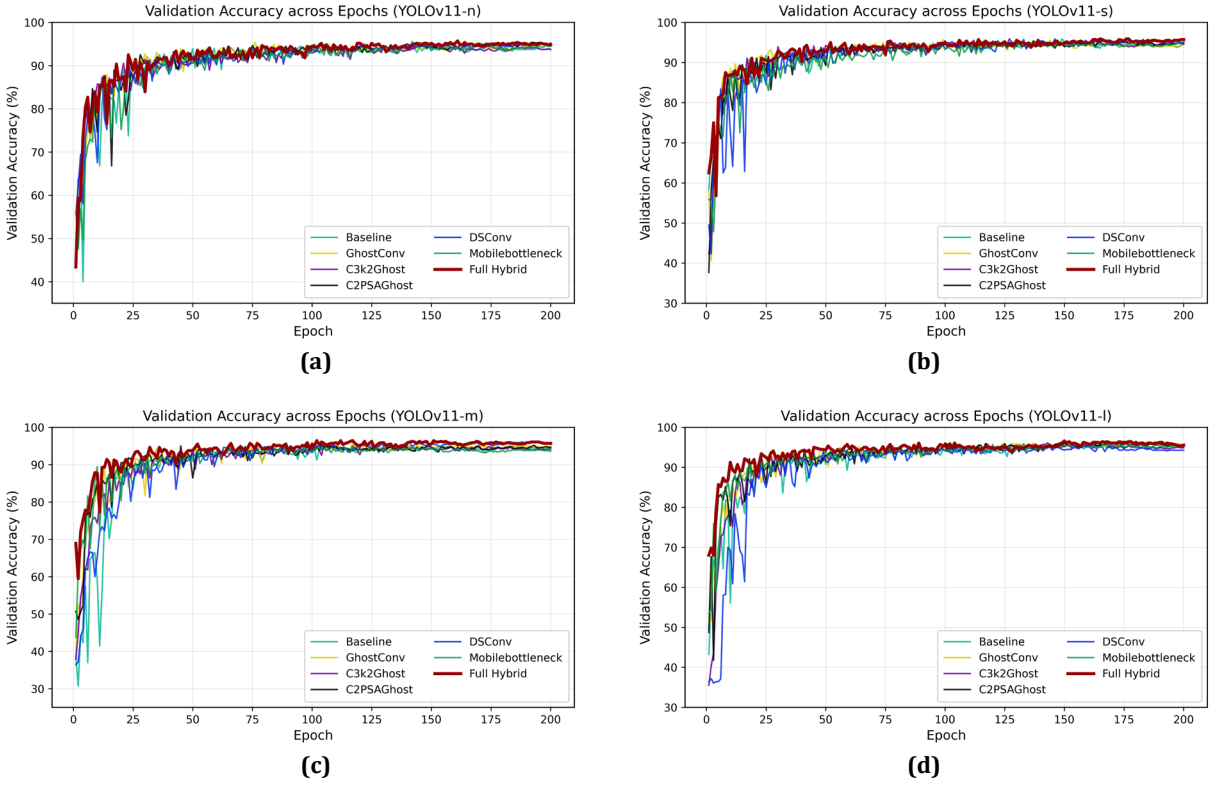
Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 490-509

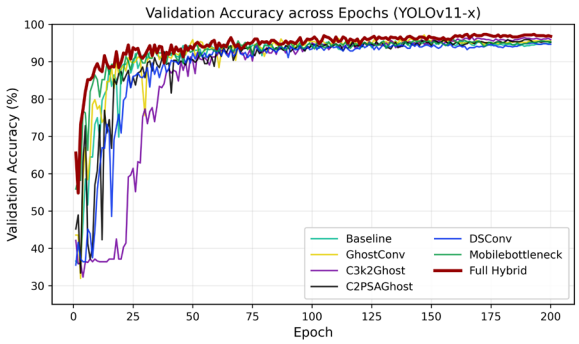
improvements in feature representation efficiency are crucial for model performance. The Wilcoxon signed-rank test demonstrated that the performance gains from GhostConv are statistically significant, strengthening the credibility of the proposed Ghost-based architectural modifications. However, the comparison between GhostConv and Full Hybrid yielded no significant difference (adjusted p = 1.000), suggesting that the performance of both models is comparable. This finding is further supported by the small effect size (r = +0.092), indicating that while Full Hybrid offers improved efficiency, it does not outperform GhostConv in terms of classification accuracy.

The effect size analysis underscores the practical significance of the observed improvements, with large effect sizes observed in the comparisons between GhostConv and C3k2Ghost, C2PSAGhost, and MobileBottleneck. This suggests that the architectural modifications introduced by GhostConv significantly enhance model performance and can be considered a meaningful contribution to red-meat classification tasks.

3.3 Training Behavior

Training behavior was further analyzed through validation accuracy and validation loss curves across all YOLOv11 scales. These curves provide additional insight into convergence speed, learning stability, and optimization consistency beyond the final evaluation metrics. The validation curves in Fig. 10 and Fig. 11 show that the proposed Ghost-based architectures, particularly GhostConv and Full Hybrid, exhibit more favorable training behavior than the baseline and other lightweight variants. Both models achieve rapid early convergence, more stable validation accuracy, and smoother loss reduction across epochs, indicating improved optimization stability and more consistent feature learning. GhostConv demonstrates strong and reliable convergence across multiple scales, while Full Hybrid tends to reach a more stable regime with lower final validation loss at medium-to-large scales. These results suggest that the proposed modifications improve not only final classification performance but also the robustness and consistency of the overall learning process.



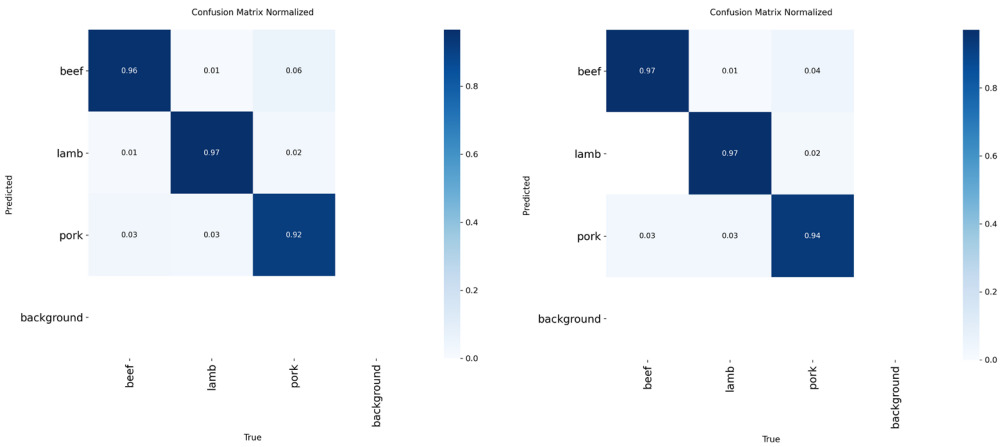


(e)

Fig. 10 *Validation accuracy across epochs. (a) Scale YOLOv11n; (b) Scale YOLOv11s; (c) Scale YOLOv11m; (d) Scale YOLOv11l; (e) Scale YOLOv11x*

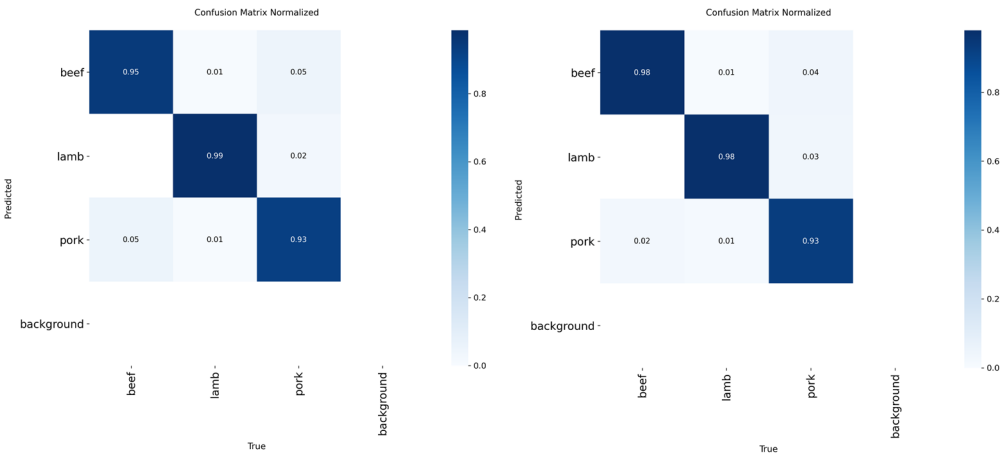
3.4 Confusion Matrix Analysis

Confusion matrix analysis was conducted to examine class-level prediction behavior and identify potential misclassification patterns among the evaluated models. By analyzing the normalized confusion matrices across all YOLOv11 scales, this analysis provides a more detailed view of how well each architecture distinguishes visually similar red-meat categories beyond the aggregate performance metrics.



(a)

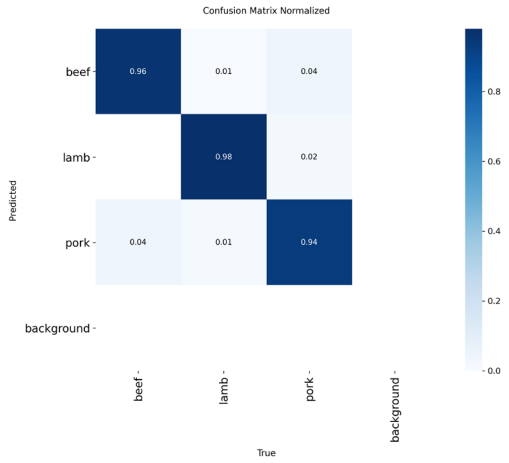
(b)



(c)

(d)

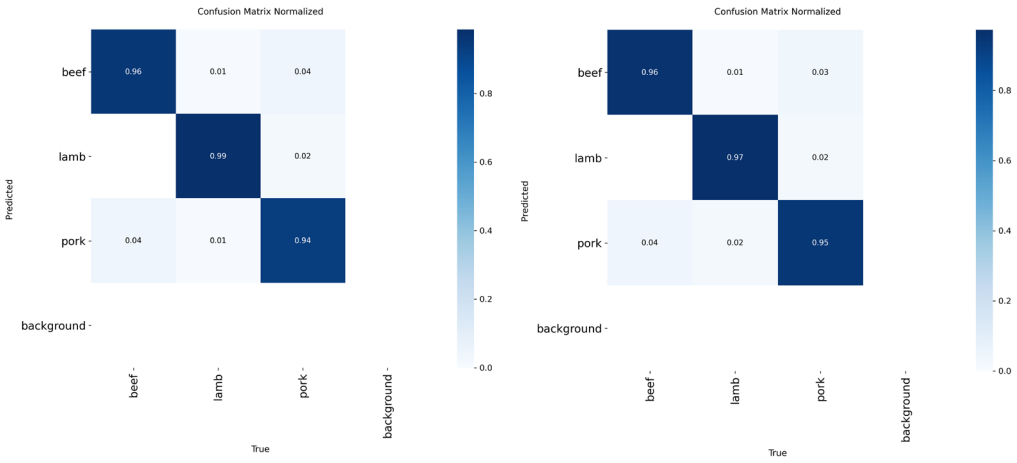
Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 490-509



(e)

Fig. 12 *Confusion matrix normalized GhostConv. (a) Scale YOLOv11n; (b) Scale YOLOv11s; (c) Scale YOLOv11m; (d) Scale YOLOv11l; (e) Scale YOLOv11x*

Figure 12 illustrates the normalized confusion matrix for the GhostConv architecture at the YOLOv11-n scale. The matrix shows a notable dominance along the diagonal, indicating that the majority of samples are correctly classified. This suggests that GhostConv effectively reduces misclassification, allowing the model to better differentiate between classes despite the visual similarities between beef, pork, and lamb. However, some minor misclassifications remain, particularly among highly similar classes, which could be attributed to challenges in distinguishing subtle differences in texture and fat distribution. In contrast, Figure 13 presents the normalized confusion matrix for the Full Hybrid architecture, which combines GhostConv, C3k2Ghost, and C2PSAGhost. Compared to GhostConv alone, this matrix shows significantly clearer class separation, especially for more challenging categories like beef and lamb. By leveraging multi-scale feature aggregation and attention-based feature selectivity, this approach allows the model to effectively distinguish finer details such as texture and color, improving the ability to discriminate between highly similar classes. The matrix also reveals strong diagonal dominance, highlighting the improved classification performance and reduced misclassifications.



(a) **(b)**

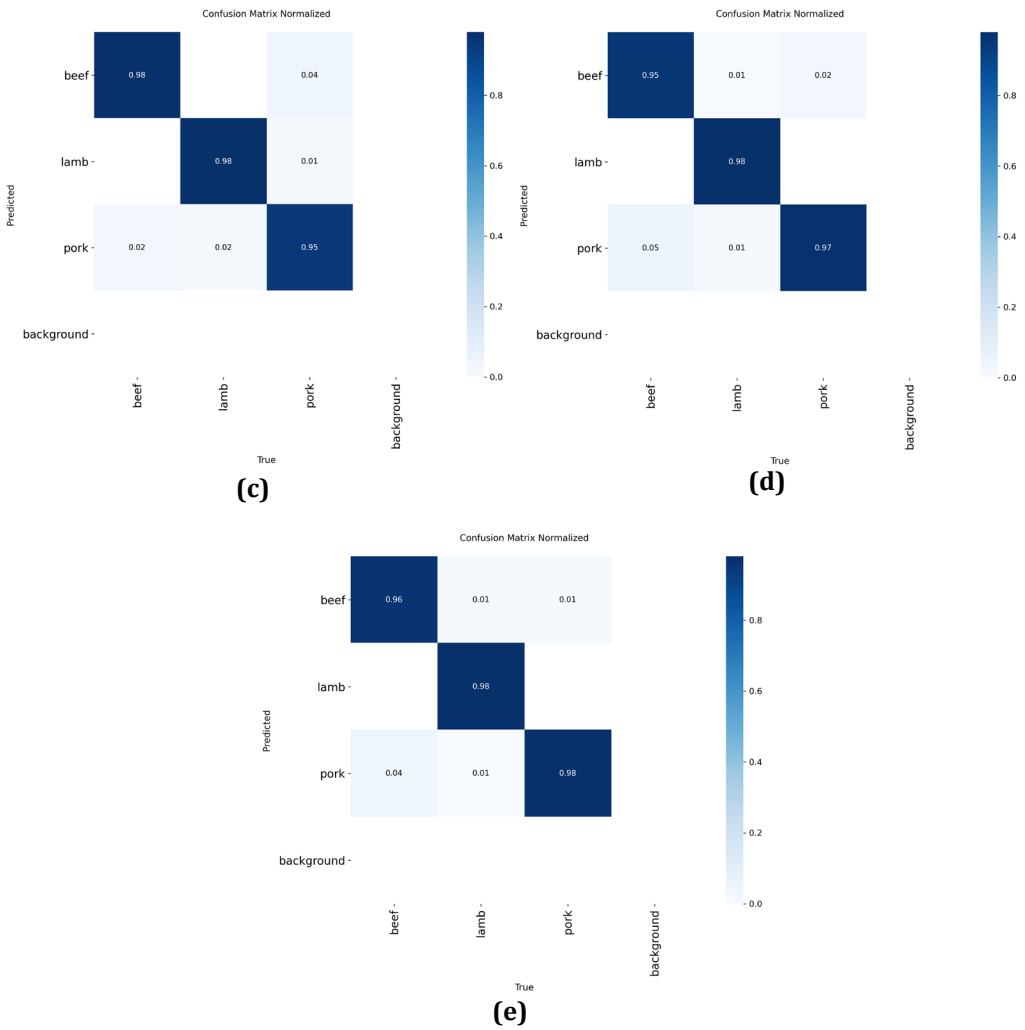


Fig. 13 *Confusion matrix normalized full hybrid. (a) Scale YOLOv11n; (b) Scale YOLOv11s; (c) Scale YOLOv11m; (d) Scale YOLOv11l; (e) Scale YOLOv11x*

Figure 14 presents the visual outputs of red-meat classification for both the GhostConv YOLOv11x and Full Hybrid YOLOv11x models. In cases where the images exhibit subtle texture variations and lighting changes, Full Hybrid consistently produces more accurate and stable predictions. This improvement indicates that Full Hybrid is better equipped to handle lighting variations and complex backgrounds, which are common challenges in red-meat classification. On the other hand, GhostConv also demonstrates its ability to improve class differentiation through channel redundancy reduction, though Full Hybrid provides even stronger results by incorporating multi-scale feature aggregation and attention-based feature selectivity, which enhance the separation of relevant features.

Int. Journal of Integrated Engineering Vol. 18 No. 3 (2026) p. 490-509

(a) (b)

Fig. 14 *Visual output of red-meat classification. (a) GhostConv YOLOv11x; (b) Full hybrid YOLOv11x*

3.5 Discussion

At The experimental results across all YOLOv11 scales indicate that performance improvements are influenced not only by network depth or model capacity, but also by the efficiency of feature representation within the architecture. In particular, the proposed GhostConv and Full Hybrid configurations demonstrate that carefully designed architectural modifications can improve classification accuracy while maintaining computational efficiency. This observation suggests that the effectiveness of the proposed approach is closely related to how redundant feature representations are suppressed and how discriminative features are selectively emphasized during the learning process.

Table 9 *Comparison with previous studies on lightweight models for red meat image classification*

Method	Architecture	Performance Metric
YOLOv5s-EGhost [11]	YOLO + Ghost	95.5%
FMEAT-Net [1]	Lightweight CNN	90.83%
MobileNetV2 Meat Classification[23]	MobileNet	94.7%
EfficientNet-B0 Meat Classification[24]	EfficientNet	95.4%
Proposed YOLOv11 GhostConv	YOLOv11	**96.60%**
Proposed YOLOv11 Full Hybrid	YOLOv11	**96.58%**

From a theoretical perspective, the stronger performance of the Full Hybrid architecture at larger YOLOv11 scales can be explained through the concepts of redundancy suppression and representation efficiency in overparameterized networks. As network capacity increases, large-scale models tend to contain substantial channel redundancy, where multiple filters learn highly correlated feature representations. The use of GhostConv helps mitigate this issue by generating informative feature maps from a reduced set of intrinsic channels, thereby reducing redundant computations while preserving discriminative information.

In addition, the integration of C3k2Ghost and C2PSAGhost enhances feature representation through complementary mechanisms. C3k2Ghost improves multi-scale feature aggregation, allowing the model to capture both fine texture patterns and broader structural cues that are important for distinguishing visually similar red-meat classes. Meanwhile, the attention-based mechanism introduced through C2PSAGhost improves feature selectivity by emphasizing semantically relevant spatial–channel responses. This combination enables the Full Hybrid architecture to utilize the representational capacity of larger YOLOv11 models more efficiently, resulting in improved class separability and more stable learning behavior.

4. Conclusion

This study demonstrates that Ghost-based architectural optimization of YOLOv11, implemented through the proposed GhostConv and Full Hybrid configurations, provides an effective solution for fine-grained red-meat classification involving beef, lamb, pork, and background classes. Across all YOLOv11 scales, the experimental

results show that performance improvements are not determined solely by network scale, but also by the efficiency of feature representation achieved through architectural modification. In particular, GhostConv consistently delivers strong classification performance across several scales, while Full Hybrid combines GhostConv, C3k2Ghost, and C2PSAGhost to improve redundancy suppression, multi-scale feature representation, and attention-based feature selectivity within a unified framework.

The advantages of the Full Hybrid design are most evident at larger YOLOv11 scales, where overparameterized models are more susceptible to feature redundancy and inefficient computation. At the YOLOv11-x scale, Full Hybrid achieves the most favorable balance between predictive performance and computational efficiency, while GhostConv attains the highest mean F1-scores on several other scales. Additional analyses based on repeated runs, effect size, training behavior, confusion matrices, and qualitative visual outputs further confirm that both proposed architectures improve convergence stability, class-level discrimination, and representational robustness. These findings indicate that the observed gains arise not merely from increased model capacity, but from more efficient and selective feature utilization within the network.

From a broader perspective, this work provides empirical and theoretical evidence that Ghost-based architectural optimization can strengthen both classification performance and computational efficiency in fine-grained visual recognition tasks. The results suggest that redundancy reduction, feature selectivity, and multi-scale feature aggregation are complementary mechanisms for improving representation quality in resource-constrained computer-vision systems. For practical applications, these findings support the use of GhostConv and Full Hybrid for image-based food inspection and red-meat authentication. Future research should extend this work through direct benchmarking against dedicated modern classification backbones, such as MobileNet, EfficientNet, ConvNeXt, or Vision Transformers, as well as through cross-dataset validation and real-time deployment analysis.

Acknowledgement

The authors would like to express their gratitude to the Intelligent Systems Laboratory (Laboratorium Sistem Cerdas), Faculty of Computer Science, Universitas Brawijaya, for providing the computational infrastructure and research environment that supported this work.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author Contribution

Author contributions are described using the CRediT (Contributor Roles Taxonomy) framework as follows. Mochamad Tono: ***Conceptualization, Methodology, Software, Formal Analysis, Investigation, Data Curation, Writing – Original Draft, Visualization****. Wayan Firdaus Mahmudy:* ***Conceptualization, Supervision, Project Administration, Writing – Review & Editing, Validation****. Rizal Setya Perdana:* ***Methodology, Validation, Writing – Review & Editing****. Muhammad Halim Natsir:* ***Resources, Domain Expertise (Animal Science), Writing – Review & Editing****.*

References

[1] M.F. Syahid, T. Haryanto, H. Rahmawan, FMEAT-Net: an efficient CNN architecture for image-based identification of beef, pork, and adulterated meat, Vis Comput 42 (2025) 83. https://doi.org/10.1007/s00371-025-04254-4.

[2] L. Wang, J. Liu, J. Pi, D. Wang, Image recognition algorithm for pork freshness based on YOLOv8n, Food and Machinery 41 (2025) 98–104. https://doi.org/10.13652/j.spjx.1003.5788.2024.80882.

[3] Y. Pang, C. Chen, Y. Yang, D. Mo, Porkolor: A deep learning framework for pork color classification, Meat Science 221 (2025) 109731. https://doi.org/10.1016/j.meatsci.2024.109731.

[4] R. Xi, X. Lyu, J. Yang, P. Lu, X. Duan, D.L. Hopkins, Y. Zhang, Non-Destructive and Real-Time Discrimination of Normal and Frozen-Thawed Beef Based on a Novel Deep Learning Model, Foods 14 (2025) 3344. https://doi.org/10.3390/foods14193344.

[5] J. Zhao, Y. Yu, S. Lu, Y. Huang, X. Yu, X. Mao, Y. Yu, Z. Li, H. Liu, Convolutional neural networks in the realm of food quality and safety evaluation: Current achievements and future prospects, Trends in Food Science & Technology 163 (2025) 105162. https://doi.org/10.1016/j.tifs.2025.105162.

[6] M. Chaman, A. El Maliki, N. Jariri, H. Dahou, H. Laâmari, A. Hadjoudja, Enhanced Deep Neural Network-Based Vehicle Detection System Using YOLOv11 for Autonomous Vehicles, in: 2025 5th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), 2025: pp. 1–6. https://doi.org/10.1109/IRASET64571.2025.11008084.

[7] C. Zheng, L. Liu, Q. Fu, Q. Yang, D. Zhang, H. Yang, YOLO-DD: a lightweight framework for UAV detection in complex environments via boundary-aware fusion, EURASIP J. Adv. Signal Process. 2025 (2025) 44. https://doi.org/10.1186/s13634-025-01253-4.

[8] H. Wang, K. Yu, Q. Li, Q. Guan, S. Gao, Lightweight medical image segmentation network based on ghost convolution and attention mechanism, in: 2023. https://doi.org/10.1117/12.2683428.

[9] C. Phoemchalard, N. Senarath, P. Malila, T. Tathong, S. Khamhan, Using orange data mining for meat classification: The preliminary application of machine learning, International Journal of Agricultural Technology 20 (2024) 2497–2512.

[10] Q. Guan, K. Yu, H. Wang, Q. Li, S. Gao, Lightweight Protective Clothing Detection Algorithm Based on Ghost Convolution and GSConv Convolution, in: 2023: pp. 73–77. https://doi.org/10.1109/ICCD59681.2023.10420656.

[11] Q. Huang, Z. Li, Improvement of YOLOv5s-Ghost model, J. Phys.: Conf. Ser. 2816 (2024) 012081. https://doi.org/10.1088/1742-6596/2816/1/012081.

[12] M.Farrel, R.Winantyo, A.Kusnadi, E.E.Surbakti, Computer Vision-Based Security System for Monitoring The Use of Personal Protective Equipment (PPE) In Workplace Involving Production Machinery, International Journal of Integrated Engineering 17 (2025) 246–257.

[13] A. Asperti, D. Evangelista, M. Marzolla, Dissecting FLOPs Along Input Dimensions for GreenAI Cost Estimations, in: Lect. Notes Comput. Sci., Springer Science and Business Media Deutschland GmbH, 2022: pp. 86–100. https://doi.org/10.1007/978-3-030-95470-3_7.

[14] J. Wei, L. Ni, L. Luo, M. Chen, M. You, Y. Sun, T. Hu, GFS-YOLO11: A Maturity Detection Model for Multi-Variety Tomato, Agronomy 14 (2024) 2644. https://doi.org/10.3390/agronomy14112644.

[15] Y. Jian, J. Yu, S. Wen, Q. Zhao, H. Hu, J. Zhang, Optimization of C2PSA Module in YOLOv11 Architecture for Semiconductor Wafer Inspection, in: 2025 International Conference on Advanced Mechatronic Systems (ICAMechS), 2025: pp. 232–237. https://doi.org/10.1109/ICAMechS68051.2025.11180983.

[16] J. Cao, W. Bao, H. Shang, M. Yuan, Q. Cheng, GCL-YOLO: A GhostConv-Based Lightweight YOLO Network for UAV Small Object Detection, Remote Sensing 15 (2023) 4932. https://doi.org/10.3390/rs15204932.

[17] D. Luo, G. Cai, Y. Lan, Research on Driving Behavior Detection Algorithm Based on YOLOv10, in: Proceedings of the 2024 8th International Conference on Electronic Information Technology and Computer Engineering, Association for Computing Machinery, New York, NY, USA, 2025: pp. 1392–1397. https://doi.org/10.1145/3711129.3711362.

[18] M. Mahala, M. Pattanaik, G. Pandey, Advancing Real-Time Object Detection for Autonomous Vehicles with YOLOv11, in: 2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), 2025: pp. 1–6. https://doi.org/10.1109/IATMSI64286.2025.10984727.

[19] D.S.B. Naik, Y.G. Bala, G.G. Krishna, T.B. Surendhra, Y. Saketh, Vehicle Counting for Traffic Analysis: A Comparative Study Using YOLOv5, YOLOv8, and YOLO11, in: J.C. Bansal, P. Jamwal, S. Hussain (Eds.), Proceedings of the International Conference on AI and Robotics, Springer Nature Switzerland, Cham, 2026: pp. 157–168. https://doi.org/10.1007/978-3-032-05545-3_13.

[20] D. Upadhyay, M. Gupta, D. Yadav, A. Upadhyay, Automated Classification of Chest CT Scans for Accurate Diagnosis and Risk Stratification: A Deep Learning Approach, in: Int. Conf. Comput., Electron., Electr. Eng. Their Appl., IC2E3, Institute of Electrical and Electronics Engineers Inc., 2024. https://doi.org/10.1109/IC2E362166.2024.10827252.

[21] M.H. Natsir, W.F. Mahmudy, M. Tono, Y.F. Nuningtyas, Advancements in artificial intelligence and machine learning for poultry farming: Applications, challenges, and future prospects, Smart Agricultural Technology 12 (2025) 101307. https://doi.org/10.1016/j.atech.2025.101307.

[22] M.S.K. Namana, B.U. Kumar, Optimizing YOLO Models for Efficient Object Detection for Surveillance Applications, in: 2025 3rd International Conference on Inventive Computing and Informatics (ICICI), 2025: pp. 1333–1338. https://doi.org/10.1109/ICICI65870.2025.11069753.

[23] A.F.A. Rahman, S.Z.M. Muji, C.U.E. Kan, Beef and Pork Meat Classification using MobileNet V2 Algorithm via Android Smartphone Application, Evolution in Electrical and Electronic Engineering 5 (2024) 292–297.

[24] A.T. Akbar, S. Saifullah, H. Prapcoyo, R. Husaini, B.M. Akbar, EfficientNet B0-Based RLDA for Beef and Pork Image Classification, in: Atlantis Press, 2024: pp. 136–145. https://doi.org/10.2991/978-94-6463-366-5_13.